

FILEID**CHMENDWRD

H 1

CCCCCCCC HH HH MM MM EEEEEEEEEE NN NN DDDDDDDD WW WW RRRRRRRR DDDDDDDD
CCCCCCCC HH HH MM MM EEEEEEEEEE NN NN DDDDDDDD WW WW RRRRRRRR DDDDDDDD
CC HH HH MMMM MMMM EE NN NN DD DD WW WW RR RR DD DD
CC HH HH MMMM MMMM EE NN NN DD DD WW WW RR RR DD DD
CC HH HH MM MM EE NNNN NN DD DD WW WW RR RR DD DD
CC HH HH MM MM EE NNNN NN DD DD WW WW RR RR DD DD
CC HHHHHHHHHHHH MM MM EEEEEEEE NN NN DD DD WW WW RRRRRRRR DD DD
CC HHHHHHHHHHHH MM MM EEEEEEEE NN NN DD DD WW WW RRRRRRRR DD DD
CC HH HH MM MM EE NN NNNN DD DD WW WW RR RR DD DD
CC HH HH MM MM EE NN NNNN DD DD WW WW RR RR DD DD
CC HH HH MM MM EE NN NN DD DD WW WW RR RR DD DD
CC HH HH MM MM EE NN NN DD DD WW WW RR RR DD DD
CCCCCCCC HH HH MM MM EEEEEEEEEE NN NN DDDDDDDD WW WW RR RR DDDDDDDD
CCCCCCCC HH HH MM MM EEEEEEEEEE NN NN DDDDDDDD WW WW RR RR DDDDDDDD

LL IIIIII SSSSSSS
LL IIIIII SSSSSSS
LL II SS SSSSSSS
LLLLLLLLL IIIIII SSSSSSS
LLLLLLLLL IIIIII SSSSSSS

ED
VO

0001 0 XTITLE 'EDTSCHMENDWRD - find the end of a word'
0002 0 MODULE EDTSCHMENDWRD (! Find the end of a word
0003 0 IDENT = 'V04-000'
0004 0) = ! File: CHMENDWRD.BLI Edit: JBS1005
0005 1 BEGIN
0006 1
0007 1 *****
0008 1 *
0009 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0010 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0011 1 * ALL RIGHTS RESERVED.
0012 1 *
0013 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0014 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0015 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0016 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0017 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0018 1 * TRANSFERRED.
0019 1 *
0020 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0021 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0022 1 * CORPORATION.
0023 1 *
0024 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0025 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0026 1 *
0027 1 *
0028 1 *****
0029 1
0030 1
0031 1 **
0032 1 FACILITY: EDT -- The DEC Standard Editor
0033 1
0034 1 ABSTRACT:
0035 1
0036 1 This module finds the end of the word entity.
0037 1
0038 1 ENVIRONMENT: Runs at any access mode - AST reentrant
0039 1
0040 1 AUTHOR: Bob Kushlis, CREATION DATE: Unknown
0041 1
0042 1 MODIFIED BY:
0043 1
0044 1 1-001 - Original. DJS 04-Feb-1981. This module was created by
0045 1 extracting the routine EDT\$SFND_EWD from module CHANGE.BLI.
0046 1 1-002 - Regularize headers. JBS 27-Feb-1981
0047 1 1-003 - Fix module name. JBS 02-Mar-1981
0048 1 1-004 - Add handling for new type of word. STS 23-Oct-1981
0049 1 1-005 - Put WPS support under a conditional. JBS 10-Feb-1983
0050 1 '--
0051 1

```
53 0052 1 %SBTTL 'Declarations'
54 0053 1
55 0054 1 TABLE OF CONTENTS:
56 0055 1 !
57 0056 1
58 0057 1 REQUIRE 'EDTSRC:TRAROUNAM';
59 0496 1
60 0497 1 FORWARD ROUTINE
61 0498 1 EDT$SFND_EWD;                                ! Find the end of a word
62 0499 1
63 0500 1 !
64 0501 1 INCLUDE FILES:
65 0502 1 !
66 0503 1
67 0504 1 REQUIRE 'EDTSRC:EDTREQ';
68 0639 1
69 0640 1 LIBRARY 'EDTSRC:SUPPORTS';
70 0641 1
71 0642 1
72 0643 1 MACROS:
73 0644 1
74 0645 1     NONE
75 0646 1
76 0647 1 EQUATED SYMBOLS:
77 0648 1
78 0649 1     NONE
79 0650 1
80 0651 1 OWN STORAGE:
81 0652 1
82 0653 1     NONE
83 0654 1
84 0655 1 EXTERNAL REFERENCES:
85 0656 1
86 0657 1     In the routine
```

```
88 0658 1 %SBTTL 'EDT$SFND_EWD - find the end of a word'  
89 0659 1  
90 0660 1 GLOBAL ROUTINE EDT$SFND_EWD ! Find the end of a word  
91 0661 1 =  
92 0662 1  
93 0663 1 ++  
94 0664 1 FUNCTIONAL DESCRIPTION:  
95 0665 1  
96 0666 1 Find the end of a word.  
97 0667 1  
98 0668 1 The number of characters between the cursor and the end of the word  
99 0669 1 is returned.  
100 0670 1  
101 0671 1 FORMAL PARAMETERS:  
102 0672 1  
103 0673 1 NONE  
104 0674 1  
105 0675 1 IMPLICIT INPUTS:  
106 0676 1  
107 0677 1 EDT$SA_US_ENT  
108 0678 1 EDT$ST_LN_BUF  
109 0679 1 EDT$SG_WRDTYPE  
110 0680 1 EDT$SA_LN_PTR  
111 0681 1  
112 0682 1 IMPLICIT OUTPUTS:  
113 0683 1  
114 0684 1 NONE  
115 0685 1  
116 0686 1 ROUTINE VALUE  
117 0687 1 The number of characters between the cursor and the end of the word.  
118 0688 1  
119 0689 1  
120 0690 1 SIDE EFFECTS:  
121 0691 1  
122 0692 1 NONE  
123 0693 1  
124 0694 1 --  
125 0695 1  
126 0696 2 BEGIN  
127 0697 2  
128 0698 2 EXTERNAL ROUTINE  
129 0699 2 EDT$ST_ENTDELIM, ! Determine if the current character is an entity delimiter  
130 0700 2 EDT$SCS_RIGHT; ! Move right a character  
131 0701 2  
132 0702 2 EXTERNAL  
133 0703 2 EDT$SA_US_ENT : VECTOR, ! Pointers to user defined entities  
134 0704 2 EDT$ST_LN_BUF, ! Current line buffer  
135 0705 2  
136 L 0706 2 %IF SUPPORT_WPS  
137 0707 2 %THEN EDT$SG_WRDTYPE, ! indicates whether delimiters are words  
138 0708 2  
139 0709 2 %FI  
140 0710 2  
141 0711 2 EDT$SA_LN_PTR; ! Current character pointer  
142 0712 2  
143 0713 2 LOCAL  
144 0714 2 SIZE:
```

```
145 0715 2
146 0716 2      SIZE = 0;
147 0717 2      + Scan up to a delimiter.
148 0718 2      |-.
149 0719 2
150 0720 2
151 0721 2      WHILE ( NOT EDT$STST_ENTDELIM (.EDTSSA_US_ENT [0])) DO
152 0722 3      BEGIN
153 0723 3      IF NOT EDT$SCS_RIGHT () THEN EXITLOOP;
154 0724 3
155 0725 3      SIZE = .SIZE + 1;
156 0726 3
157 0727 2      END;
158 0728 2
159 0729 2      + If the delimiter was a space, or there were no characters before the
160 0730 2      delimiter, then scan past all trailing spaces (but do not cross a line).
161 0731 2
162 0732 2      -.
163 0733 2
164 L 0734 2      %IF SUPPORT_WPS
165 0735 2      %THEN
166 0736 2
167 0737 3      IF (.EDTSSG_WRDTYPE EQL DELIMITED)
168 0738 2      THEN
169 0739 2      %FI
170 0740 2
171 0741 3      BEGIN
172 0742 3
173 0743 4      IF ((.SIZE EQL 0) OR (CH$RCHAR (.EDTSSA_LN_PTR) EQL %C' '))
174 0744 3      THEN
175 0745 3
176 0746 3      DO
177 0747 4      BEGIN
178 0748 4      EDT$SCS_RIGHT ();
179 0749 4      SIZE = .SIZE + 1;
180 0750 4      END
181 0751 4      UNTIL (CH$PTR_EQL (.EDTSSA_LN_PTR, CH$PTR (EDT$ST_LN_BUF)) OR
182 0752 3      (CH$RCHAR (.EDTSSA_LN_PTR) NEQ %C' '));
183 0753 3
184 0754 3      END
185 0755 3
186 L 0756 3      %IF SUPPORT_WPS
187 0757 3      %THEN
188 0758 2      ELSE
189 0759 2      + Delimiters are not considered words so pass over any delimiters
190 0760 2
191 0761 2      |-.
192 0762 3      BEGIN
193 0763 3
194 0764 4      IF (.SIZE EQL 0) OR (EDT$STST_ENTDELIM (.EDTSSA_US_ENT [0]))
195 0765 3
196 0766 3
197 0767 3      DO
198 0768 4      BEGIN
199 0769 4      IF NOT EDT$SCS_RIGHT () THEN EXITLOOP;
200 0770 4
201 0771 4
```

EDT\$CHMENDWRD
V04-000

EDT\$CHMENDWRD - find the end of a word
EDT\$SFND_EWD - find the end of a word

E 2
15-Sep-1984 23:51:38
14-Sep-1984 12:22:28
VAX-11 Bliss-32 V4.0-742
[EDT.SRC]CHMENDWRD.BLI;1

Page 5
(3)

202 0772 4
203 0773 4
204 0774 3
205 0775 3
206 0776 3
207 0777 3
208 0778 3
209 0779 2
210 0780 2
211 0781 1
SIZE = .SIZE + 1;
END
UNTIL (NOT EDT\$STST_ENTDELIM (.EDTSSA_US_ENT [0]));
END
XFI
RETURN (.SIZE);
END;
! of routine EDT\$SFND_EWD

.TITLE EDT\$CHMENDWRD EDT\$CHMENDWRD - find the end of a
word

.IDENT \V04-000\

.EXTRN EDT\$STST_ENTDELIM
.EXTRN EDT\$SCS_RIGHT, EDTSSA_US_ENT
.EXTRN EDT\$STLN_BUF, EDT\$SG_WRDTYPE
.EXTRN EDTSSA_LN_PTR

.PSECT _EDT\$CODE, NOWRT, SHR, PIC,2

56 00000G00G	00 007C 00000	.ENTRY	EDT\$SFND_EWD, Save R2,R3,R4,R5,R6	0660
55 00000000G	00 9E 00002	MOVAB	EDTSSA_US_ENT, R6	
54 00000000G	00 9E 00009	MOVAB	EDTSSA_LN_PTR, R5	
53 00000000G	00 9E 00010	MOVAB	EDT\$STST_ENTDELIM, R4	
	52 D4 0001E	MOVAB	EDT\$SCS_RIGHT, R3	
	66 DD 00020	1\$: CLRL	SIZE	0716
64 0A	01 FB 00022	PUSHL	EDTSSA_US_ENT	0721
63 04	50 E8 00025	CALLS	#1, EDT\$STST_ENTDELIM	
	00 FB 00028	BLBS	R0, 2\$	
	50 E9 0002B	CALLS	#0, EDT\$SCS_RIGHT	0724
	52 D6 0002E	BLBC	R0, 2\$	
	EE 11 00030	INCL	SIZE	0726
00000000G	00 D5 00032	BRB	1\$	0721
	27 12 00038	TSTL	EDT\$SG_WRDTYPE	0737
	52 D5 0003A	BNEQ	4\$	
	08 13 0003C	TSTL	SIZE	0743
50 20	65 D0 0003E	BEQL	3\$	
	60 91 00041	MOVL	EDTSSA_LN_PTR, R0	
	31 12 00044	CMPB	(R0), #32	
63	00 FB 00046	3\$: BNEQ	7\$	
	52 D6 00049	CALLS	#0, EDT\$SCS_RIGHT	0748
51	65 D0 0004B	INCL	SIZE	0749
50 00000000G	00 9E 0004E	MOVL	EDTSSA_LN_PTR, R1	0751
50	51 D1 00055	MOVAB	EDT\$STLN_BUF, R0	
	1D 13 00058	CMPL	R1, R0	
20	61 91 0005A	BEQL	7\$	
	E7 13 0005D	CMPB	(R1), #32	0752
	16 11 0005F	BEQL	3\$	
	52 D5 00061	4\$: BRB	7\$	0737
	08 13 00063	TSTL	SIZE	0764
64 0A	66 DD 00065	5\$: BEQL	6\$	
	01 FB 00067	PUSHL	EDTSSA_US_ENT	
	50 E9 0006A	CALLS	#1, EDT\$STST_ENTDELIM	
		BLBC	R0, 7\$	

EDTSCHMENDWRD EDTSCHMENDWRD - find the end of a word
VO4-000 EDT\$SFND_EWD - find the end of a word

F 2
15-Sep-1984 23:51:38 VAX-11 Bliss-32 V4.0-742
14-Sep-1984 12:22:28 [EDT.SRC]CHMENDWRD.BLI;1

Page 6
(3)

63	00	FB 0006D	6\$: CALLS #0, EDT\$CS_RIGHT	: 0770
04	50	E9 00070	BLBC R0, 7\$: 0772
	52	D6 00073	INCL SIZE	: 0774
	EE	11 00075	BRB SS	: 0780
50	52	D0 00077	7\$: MOVL SIZE, R0	: 0781
		04 0007A	RET	

: Routine Size: 123 bytes, Routine Base: _EDT\$CODE + 0000

: 212 0782 1
: 213 0783 1 !<BLF/PAGE>

EDT\$CHMENDWRD
V04-000 EDT\$CHMENDWRD - find the end of a word
 EDT\$SFND_EWD - find the end of a word
:
: 215 0784 1 END
: 216 0785 1
: 217 0786 0 ELUDOM

G 2
15-Sep-1984 23:51:38 VAX-11 Bliss-32 V4.0-742
14-Sep-1984 12:22:28 [EDT.SRC]CHMENDWRD.BLI;1
 . of module EDT\$CHMENDWRD

Page 7
(4)

E
V

PSECT SUMMARY

Name	Bytes	Attributes
_EDT\$CODE	123	NOVEC,NOWRT, RD , EXE, SHR, LCL, REL, CON, PIC,ALIGN(2)

Library Statistics

File	-----	Symbols	-----	Pages	Processing
	Total	Loaded	Percent	Mapped	Time
-\$255\$DUA28:[EDT.SRC]EDT.L32;1	377	2	0	40	00:00.2
-\$255\$DUA28:[EDT.SRC]PSECTS.L32;1	2	1	50	7	00:00.1
-\$255\$DUA28:[EDT.SRC]SUPPORTS.L32;1	2	1	50	5	00:00.1

COMMAND QUALIFIERS

BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACEBACK/LIS=LIS\$:CHMENDWRD/OBJ=OBJ\$:CHMENDWRD MSRC\$:CHMENDWRD.BLI/UPDATE=(ENH\$:
HMENDWRD)

Size: 123 code + 0 data bytes
Run Time: 00:12.7
Elapsed Time: 00:33.2
Lines/CPU Min: 3707
Lexemes/CPU-Min: 10584
Memory Used: 82 pages
Compilation Complete

0131 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

CHMFINENT LIS
CHMCOUNT LIS CHMGINSTR LIS CHMGSUSTR LIS
CHMINIT LIS
CHMINSMOD LIS
CHMINSTAB LIS
CHMEMESS LIS
CHMENDTRM LIS
CHMGBUF LIS
CHMINSTR LIS
CHMGOIR LIS CHMGOISTR LIS CHMHLPKPD LIS
CHMINDATE LIS
CHMINSCHR LIS
CHMINSTR LIS
CHMINSTR LIS
CHMEXVERB LIS
CHMENDWRD LIS
CHMEXCOM LIS
CHMKEYRD LIS